



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/712,544

11/13/2003

Scott Carrier

RSW920030233US1 (128)

9171

46320

7590

07/24/2008

CAREY, RODRIGUEZ, GREENBERG & PAUL, LLP

STEVEN M. GREENBERG

950 PENINSULA CORPORATE CIRCLE

SUITE 3020

BOCA RATON, FL 33487

EXAMINER

PATEL, MANGLESH M

ART UNIT

PAPER NUMBER

2178

MAIL DATE

DELIVERY MODE

07/24/2008

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/712,544
Filing Date: November 13, 2003
Appellant(s): CARRIER, SCOTT

Scott D. Paul
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 4/25/2008 including the supplemental Brief filed 5/5/2008 correcting the status of the claims and appealing from the Office action mailed 1/25/2008.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

U.S. Pub 2005/0028084 A1, Dziejma, Filed Jul 27, 2004 with provisional
date of application 60/490,590 filed Jul 28, 2003.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 101

1. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. Regarding independent claim 1, the claim describes a system but fails to include any hardware elements in the system such as a CPU. Instead the claims describe the use of a system for a client device, it is unclear since the system is merely steps that may be used by a device only if it was embodied in a computer readable medium. If the system is directed to software it should be embodied inside a computer readable medium, if for hardware it should recite a hardware element in the claims such as a processor. A validation processor is not an actual hardware element. Furthermore a system that can be used for client devices is not sufficient because the system itself is not tangibly embodied in a computer readable medium to be used by anything.

Appropriate corrections are required.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1, 6, 11 and 15 remain rejected under 35 U.S.C. 103(a) as being unpatentable over Dziejma (U.S. Pub 2005/0028084, filed Jul 27, 2004, with a valid priority date of Jul 28, 2003) in view of Sokolov (U.S. 6,823,504, filed Nov 15, 2000).

Regarding Independent claim 1, A lightweight pattern validation system for a client device receiving markup defining a form, comprising: a validation processor separate from said markup and configured with a prototype interface for receiving both a field validation pattern and also form based input to be validated against said field validation pattern; a validation script library within said client device and packaging said validation processor, wherein the form has at least one form based input field programmed for validation using said validation processor; a library reference to said script library disposed in said markup; a function call to said validation processor further disposed in said markup, said function call having a configuration for passing a reference to a value in said at least one form based input field for validation in said validation processor; a plurality of additional function calls to said validation processor disposed in said markup, each additional one of said functional calls having a configuration for passing a reference to a value in a corresponding form based input field for validation in

said validation processor; and a validation shell function encapsulating said function calls.

Dziejma teaches a form field validation engine which is separate from the markup and resides on the client device. Furthermore the engine that handles the validation includes scripts defined by the FVE (form validation engine) code, which includes several scripting functions for validation of each field of a form in the markup. Furthermore all is done on the client device. Also the markup field form includes various standard including Xforms and includes script reference, which is function calls to the FVE. The markers reference such functions in the markup. The form includes an interface for collecting data. (See abstract, fig 3, fig 6, fig 8, paragraphs 5-8, 9-12, 40-41 & appendix A). Although Dziejma teaches the use of JavaScript in the FVE, he only shows function calls defined within the engine and fails to show reference to a separate library objects referenced by JavaScript. However Sokolov explicitly teaches the use of libraries which are interfaced with JavaScript, such interfacing includes a shell function that encapsulates the function calls thereby allowing access to the library of functions (see abstract & column 21). Thus at the time of the invention it would have been obvious to the skilled artisan to have modified the script definitions of Dziejma to include reference to various JavaScript libraries has taught by Sokolov. The motivation for doing so would have been to provide extensibility to the validation engine by referencing libraries of scripting objects in JavaScript

without constantly accessing a server, thus improving form validation on client devices.

Regarding Independent claim 6, A pattern validation method comprising the steps of: retrieving a value for a form based input field from a form defined in markup rendered in a content browser; passing said retrieved value along with a validation pattern for said form based input field to a validation process disposed within a lightweight validation library separate from and coupled to said rendered markup; and, validating said retrieved value according to said validation pattern in said content browser; repeating said retrieving, passing and validating steps for at least one additional value for at least one additional form based input field disposed in said markup rendered in said content browser; and performing said retrieving, passing, validating and repeating steps in a validation shell function disposed in said markup rendered in said content browser.

Dziejma teaches a form field validation engine which is separate from the markup and resides on the client device. Furthermore the engine that handles the validation includes scripts defined by the FVE (form validation engine) code, which includes several scripting functions for validation of each field of a form in the markup. Furthermore all is done on the client device. Also the markup field form includes various standard including Xforms and includes script reference, which is function calls to the FVE. The markers reference such functions in the

markup. The form includes an interface for collecting data. (See abstract, fig 3, fig 6, fig 8, paragraphs 5-8, 9-12, 40-41 & appendix A). Furthermore he teaches retrieving, passing and validating steps defined in the FVE shown in appendix A using conditional statements. Although Dziejma teaches the use of JavaScript in the FVE, he only shows function calls defined within the engine and fails to show reference to a separate library objects referenced by JavaScript. However Sokolov explicitly teaches the use of libraries which are interfaced with JavaScript, such interfacing includes a shell function that encapsulates the function calls thereby allowing access to the library of functions (see abstract & column 21). Thus at the time of the invention it would have been obvious to the skilled artisan to have modified the script definitions of Dziejma to include reference to various JavaScript libraries has taught by Sokolov. The motivation for doing so would have been to provide extensibility to the validation engine by referencing libraries of scripting objects in JavaScript without constantly accessing a server, thus improving form validation on client devices.

Regarding Independent claim 11, A machine readable storage having stored thereon a computer program for pattern validation, the computer program comprising a routine set of instructions which when executed by the machine cause the machine to perform the steps of: retrieving a value for a form based input field from a form defined in markup rendered in a content browser; passing said retrieved value along with a validation pattern for said form based input field

to a validation process disposed within a lightweight validation library separate from and coupled to said rendered markup; validating said retrieved value according to said validation pattern in said content browser; repeating said retrieving, passing and validating steps for at least one additional value for at least one additional form based input field disposed in said markup rendered in said content browser; and performing said retrieving, passing, validating and repeating steps in a validation shell function disposed in said markup rendered in said content browser.

Dziejma teaches a form field validation engine which is separate from the markup and resides on the client device. Furthermore the engine that handles the validation includes scripts defined by the FVE (form validation engine) code, which includes several scripting functions for validation of each field of a form in the markup. Furthermore all is done on the client device. Also the markup field form includes various standard including Xforms and includes script reference, which is function calls to the FVE. The markers reference such functions in the markup. The form includes an interface for collecting data. (See abstract, fig 3, fig 6, fig 8, paragraphs 5-8, 9-12, 40-41 & appendix A). Furthermore he teaches retrieving, passing and validating steps defined in the FVE shown in appendix A using conditional statements. Although Dziejma teaches the use of JavaScript in the FVE, he only shows function calls defined within the engine and fails to show reference to a separate library objects referenced by JavaScript. However

Sokolov explicitly teaches the use of libraries which are interfaced with JavaScript (see abstract). Thus at the time of the invention it would have been obvious to the skilled artisan to have modified the script definitions of Dziejma to include reference to various JavaScript libraries has taught by Sokolov. The motivation for doing so would have been to provide extensibility to the validation engine by referencing libraries of scripting objects in JavaScript without constantly accessing a server, thus improving form validation on client devices.

Regarding Dependent claim 15, with dependency of claim 1, Dziejma wherein the client device is a pervasive device (see fig 1).

*It is noted that any citation **[[s]]** to specific, pages, columns, lines, or figures in the prior art references and any interpretation of the references should not be considered to be limiting in any way. A reference is relevant for all it contains and may be relied upon for all that it would have reasonably suggested to one having ordinary skill in the art. **[[See, MPEP 2123]]***

(10) Response to Argument

(note: The following application was received by the new Examiner of record: Manglesh Patel on 4/16/2007, all future correspondence should be updated accordingly).

(1) Appellant Argues:

Appellant respectfully submit that the Examiner's analysis is fatally flawed. As is readily apparent from Appellant's disclosure, a validation processor could require the use of a computer device. Thus the claimed invention is directed to statutory subject matter. (pg 6, paragraphs 1-4)

The Examiner Respectfully Disagrees: Contrary to arguments made by Appellant, use of the word “system” does not inherently mean that the claim is directed to a machine. Only if at least one of the claimed elements of the system is a physical part of a device can the system constitute part of a device or a combination of devices to be a machine within the meaning of 101. A validation processor is not an actual hardware element; instead it describes software per se thus failing to fall within the statutory category of invention because it fails to be tangibly embodied in a computer readable medium to be used by anything including “a client device”.

(2) Appellant Argues:

Appellant has compared the statement of the rejection found on pages 2-6 of the Fifth Office Action with the statement of the rejection found on pages 3-12 of the Fourth Office Action. Upon making this comparison, Appellant has been unable to discover any substantial differences between the respective statements of the rejection. (pg 7, paragraph 4)

Appellant should not be placed in a position to guess the basis of the examiners rejection. (pg 13, paragraph 2)

The Examiner Respectfully Disagrees: The Final office action dated 1/25/2008 has rejected all pending claims and further provided an explanation with citations for all the rejections of each claim as support. Furthermore it has been cited in numerous actions that the teachings of a reference are not limited to specific portions, the reference as a whole must be considered by the appellant. Nonetheless appellant has decided to attack the format/structure used by the examiner to reject the claimed subject matter instead of the underlying rejection itself. Appellant has failed to “**clearly designate**” which limitations are disagreed upon to disprove the teachings nor show any evidence regarding the differences between the claimed subject matter and the cited references. The Examiner will provide a mapping to each limitation of Independent claim 1 using the previous explanations to assist the Appellant.

Claims 1 : A lightweight pattern validation system for a client device receiving markup defining a form, comprising: a validation processor separate from said markup and configured with a prototype interface for receiving both a field validation pattern and also form based input to be validated against said field validation pattern; a validation script library within said client device and packaging said validation processor, wherein the form has at least one form based input field programmed for validation using said validation processor (See abstract, fig 3, fig 6, fig 8, paragraphs 5-8, 9-12, 40-41 & appendix A, Dziejma teaches a form field validation engine which is separate from the markup and resides on the client device. Furthermore the engine that handles the validation includes scripts defined by the FVE (form validation engine) code, which includes several scripting functions for validation of each field of a form in the markup. Furthermore all is done on the client device. Also the markup field form includes script reference, which is function calls to the FVE. The markers reference such functions in the markup. The form includes an interface for collecting data.); Dziejma discloses a function call to said validation processor further disposed in said markup, said function call having a configuration for passing a reference to a value in said at least one form based input field for validation in said validation processor (appendix A, Dziejma discloses several function calls for the form validation engine within the markup document thereby passing reference values to the input field for validation by the FVE); Although Dziejma teaches the use of JavaScript in the FVE, he only shows function calls defined within the engine and

fails to show reference to a separate library objects referenced by JavaScript. However Sokolov explicitly teaches a library reference to said script library disposed in said markup (see abstract & column 21, Sokolov discloses a plurality of additional function calls to said validation processor disposed in said markup, each additional one of said functional calls having a configuration for passing a reference to a value in a corresponding form based input field for validation in said validation processor; and a validation shell function encapsulating said function calls (see abstract & column 21, Sokolov discloses the use of libraries which are interfaced with JavaScript, such interfacing includes a shell function that encapsulates the function calls thereby allowing access to the library of functions). Thus at the time of the invention it would have been obvious to the skilled artisan to have modified the script definitions of Dziejma to include reference to various JavaScript libraries has taught by Sokolov. The motivation for doing so would have been to provide extensibility to the validation engine by referencing libraries of scripting objects in JavaScript without constantly accessing a server, thus improving form validation on client devices.

(3) Appellant Argues:

Although the '084 Publication claims priority to a date of July 28, 2003, based upon U.S. Provisional Application No. 60/490,590, this claiming of

priority is not dispositive as to whether or not all the teachings found in the '084 Publication are found in the '590 Provisional. (pg 11, paragraph 1)

The Examiner Respectfully Disagrees: The Examiner has already presented a prima facie case of obviousness and designated, as nearly as practicable, the particular part being relied upon in the rejection by using specific citations of the '218 application. Furthermore the examiner prior to using the provisional date relied upon in the '590 application has already reviewed and determined that "the provisional application properly supports the subject matter relied upon...". Once again the Appellant has failed to **clearly designate** or specifically show which portions used in the rejection are not supported in the '590 provisional application, despite having access to Public Pair. Nonetheless the Examiner will provide the appropriate mappings between the subject matter of the '218 and '590 applications to assist both the Board and the Appellant.

The specific sections relied on in the rejection using the '218 application includes: abstract, fig 3, fig 6, fig 8, paragraphs 5-8, 9-12, 40-41 & appendix A. Specifically the portion relied upon include the engine that handles the validation includes scripts defined by the FVE (form validation engine) code, which includes several scripting functions for validation of each field of a form in the markup ('590 Provisional: see pg 1, paragraph 1 describing the FVE & see pgs 4-8 describing the FVE using JavaScript with associated function for validation). Also the

markup field form includes script reference, which is a function calls to the FVE.

The markers reference such functions in the markup ('590 Provisional: see page 2, which states "The FVE takes as input a pointer to an HTML form. Embedded in the field definition of the form are three new markers. The valid marker

specifies what kind of validation is to be performed on the field"). The form

includes an interface for collecting data ('590 Provisional: see figs 1-4, showing the interface for collecting data in a client/server fashion). Furthermore he

teaches retrieving, passing and validating steps defined in the FVE shown in

appendix A using conditional statements ('590 Provisional: see FVE code pg 4 retrieving and passing values such as formName of the function Validate

including the conditional statements defined by the If..Else statements on pg 4 -

5). Function calls encapsulated within a validation shell of markup ('590

Provisional: see FVE code pg 4-8 teachings several functional calls encapsulated in a validation shell which is within the FVE Code).

(3) Appellant Argues:

Specifically, independent claims 1, 6, and 11 each include the concept that the functional calls are encapsulated within a validation shell of markup.

(pg 14, paragraph 2)

The Examiner Respectfully Disagrees: The FVE (field validation engine, see abstract) of Dziejma represents a shell that describes several validation function calls encapsulated within the underlying HTML document (see appendix A). Function calls encapsulated within a validation shell of markup ('590 Provisional: see FVE code pg 4-8 teachings several functional calls encapsulated in a validation shell which is within the FVE Code).

(4) Appellant Argues:

However, Dziejma does not teach that the validation engine requires additional access to the server for extensibility. Instead, it appears that the validation engine of Dziejma, when originally received from the server, includes all the necessary functions and does not have to go back to the server. Thus, the problem allegedly solved by Sokolov is already addressed by Dziejma, and based upon common sense, one having ordinary skill in the art would not look to solve a problem that is already solved. (pg 15, paragraph 2)

The Examiner Respectfully Disagrees: Dziejma paragraph 40 states "The described **form validation method may be also used on the server for performing server-side validation**. In that case the **form validation engine resides in the server** and the form with the embedded markers and data is

submitted to the server either locally or via the network connection.” Although Dziejma teaches the use of JavaScript in the FVE, he only shows function calls defined within the engine and fails to show reference to a separate library objects referenced by JavaScript. However Sokolov explicitly teaches the use of libraries which are interfaced with JavaScript (see abstract). Thus at the time of the invention it would have been obvious to the skilled artisan to have modified the script definitions of Dziejma to include reference to various JavaScript libraries has taught by Sokolov to provide extensibility to the field validation engine of Dziejma.

(5) Appellant Argues:

Instead Dziejma merely describes a server 410 and a client 420, which is not the disclosure of a pervasive device. (pg17, paragraph 2)

The Examiner Respectfully Disagrees: Appellant has once again failed to provide evidence to show which sections of the specification in detail define the term “**Pervasive** device”, instead relying on his own opinion. The Examiner however provides the general definition of the term Pervasive to give the claim broadest reasonable interpretation.

Google: Definition of : Pervasive: **Manifested throughout**; pervading, permeating, penetrating or affecting everything

Thus the teachings of Dziejma have already established a client/server software architecture as that well known in the art. Since Dziejma supports both server side and client side validation as recited in paragraph 40 he describes a pervasive device (client device), since a client device is manifested throughout a typical distributed system. Furthermore since the access to the server from the users client machine shown in fig 1 and described in paragraph 22 is done accessing a URL, it would have been obvious for the skilled artisan to have used the client device (Specifically including a PDA/mobile device) to access a URL, since Sokolov deals with markup documents (see column 1, lines 55-67 of Sokolov).

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

Art Unit: 2176

The Examiner appreciates appellant's effort, however for the above reasons; it is believed that the rejection should be sustained.

Respectfully submitted,

Manglesh M Patel
/Manglesh M Patel/
Patent Examiner (AU 2178)

07/17/2008

Conferees:

/CESAR B PAULA/
Primary Examiner, Art Unit 2178

/Stephen S. Hong/
Supervisory Patent Examiner, Art Unit 2178
Stephen S. Hong
Supervisory patent Examiner
Art Unit 2178

/Doug Hutton/
Doug Hutton
Supervisory Primary Examiner
Technology Center 2100

Doug Hutton
Supervisory Patent Examiner
Art Unit 2176

Application/Control Number: 10/712,544
Art Unit: 2176

Page 20